Objectives

- More on using strings
- Computer's representations of data types

Strings

- Actually a sequence of characters
  - Example:
    ```
    string = "The Beatles"
    ```

- Length of the string: 11
  - Built-in function: `len(string)` to find length of a string

Iterating Through a String

- For each character in the string
  ```
  for char in string:
      print char
  ```
  ```
  'T' 'h' 'e' ' ' 'B' 'e' 'a' 't' 'l' 'e' 's'
  ```
  - An integer
    ```
    for pos in xrange(len(string)):
        print string[pos]
    ```
    ```
    0 1 2 3 4 5 6 7 8 9 10
    ```
  - For each position in the string

Substrings Operator []

- Look at a particular character in the string
  ```
  Syntax: str[<integer expression>]
  ```
  - [Positive values]: index of character
  - [Negative values]: count backwards from end

- Look at a sequence of characters in the string
  ```
  Syntax: str[start:end]
  ```

Testing for Substrings

- Using the in operator
  ```
  Used in before in for loops
  ```
  ```
  Syntax:
  ```
  substring in string
  ```
  ```
  Evaluates to True or False
  ```
  ```
  Example:
  ```
  if "cat" in name:
      print name, "contains 'cat"
  ```

Representations of Data

- Computer needs ways to represent different types of data
  ```
  Eventually, all boils down to 1s and 0s
  ```
- Computer needs to translate between what humans know (decimal, strings) to what computer knows (binary) and back again
  ```
  ```
  Seems like a divergence on strings but just wait...
Decimal Representations

- Decimal is base 10
- Digits: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
- Each position in a decimal number represents a power of 10

Example: 54,087

\[5 \times 10^4 + 4 \times 10^3 + 0 \times 10^2 + 8 \times 10^1 + 7 \times 10^0\]

Number Representations

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Decimal</th>
<th>Binary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base</td>
<td>10</td>
<td>2</td>
</tr>
<tr>
<td>Digits</td>
<td>0, 1, 2, 3, 4, 5, 6, 7, 8, 9</td>
<td>0, 1</td>
</tr>
<tr>
<td>Position represents</td>
<td>Power of 10</td>
<td>Power of 2</td>
</tr>
</tbody>
</table>

- Binary: two values (0, 1)
  - Like a light switch (either off or on) or booleans (either True or False)
- 0 and 1 are binary digits or bits
  - 64-bit machine: represents numbers (and other data) with 64 bits

Binary Representation

- Binary number: 1101

\[1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0\]

Practice: what is the decimal value of the binary number 10110?

Algorithm: Converting Binary to Decimal

- Accumulator design pattern
- Read in the binary number as a string
  - The starting exponent will be the length of the string - 1
- Initialize the result to zero
- For each bit in the binary number
  - Multiply the bit by the appropriate power of 2
  - Add this to the result
  - Reduce the exponent by 1
- Print the result

Implement algorithm binaryToDecimal.py
Algorithm: Converting Decimal to Binary

- Read in the decimal as an integer
- Initialize the result to the empty string
- Repeat until the decimal is 0:
  - result = str(decimal % 2) + result
  - decimal = decimal / 2
- Print the result

Try out algorithm with 22
Implement algorithm decimalToBinary.py

String Representations

- A string is a sequence of characters
- Each character is stored as a binary number
- ASCII (American Standard Code for Information Interchange) is one standard encoding for characters
  - A problem with ASCII is that it is based on the English language
  - Cannot represent other types of characters
- Unicode is a new standard

ASCII Questions

- Lowercase letters are represented by what range of numbers?
- Uppercase letters are represented by what range of numbers?
- What is the difference between the decimal encoding of 'M' and 'N'? Between 'm' and 'n'? 1

Translating to/from ASCII

- Translate a character into its ASCII numeric code using built-in function ord
  - ord('a') == 97
- Translate an ASCII numeric code into its character using built-in function chr
  - chr(97) == 'a'

This Week

- Broader Issue: Excel Bug
- Lab 5
  - Graphics, animation
  - Text processing